1. Write a Java program to connect to a MySQL database using JDBC.

```
package Assesement_day11;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class connect_mySQL_database {

public static void main(String[] args) {
String dbUrl = "jdbc:mysql://localhost:3306/mydb";
String dbUser = "root";
String dbPassword = "root";

try {
Class.forName("com.mysql.cj.jdbc.Driver");

Connection conn = DriverManager.getConnection(dbUrl,
dbUser, dbPassword);

Statement stmt = conn.createStatement();

ResultSet rs = stmt.executeQuery("SELECT * FROM students");

while (rs.next()) {
System.out.println(rs.getString(1) + " " + rs.getString(2));
}
```

```
conn.close();
} catch (ClassNotFoundException e) {
System.out.println("MySQL JDBC driver not found.");
} catch (SQLException e) {
System.out.println("Error connecting to the database or
executing query: " + e.getMessage());
}

}

}
```

Output:

```
1 Dhana
2 Sri
3 Sanjana
4 Penugonda
```

2. Create a Java class to insert student records into a database table.

```
package Assesement_day11;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.SQLException;

import java.util.Scanner;


public class insert_student_records {
```

```
public static void main(String[] args) {
        try {

                Class.forName("com.mysql.cj.jdbc.Driver");


                String url =
"jdbc:mysql://localhost:3306/mydb";


                String username = "root";
                String password = "root";


                Connection con =
DriverManager.getConnection(url, username, password);
                Scanner scanner = new
Scanner(System.in);
                System.out.print("Enter student ID: ");
                int id = scanner.nextInt();
                System.out.print("Enter student name: ");
                String name = scanner.next();
```

```
                System.out.print("Enter student age: ");
                int age = scanner.nextInt();
                String query = "INSERT INTO student (id,
name, age) VALUES (?, ?, ?)";
                PreparedStatement pstmt =
con.prepareStatement(query);
                pstmt.setInt(1, id);
                pstmt.setString(2, name);
                pstmt.setInt(3, age);
                int rowsAffected = pstmt.executeUpdate();
                if (rowsAffected > 0) {
                    System.out.println("Student record
inserted successfully");
                } else {
                    System.out.println("Failed to insert
student record");
                }
                con.close();
            } catch (ClassNotFoundException e) {
                System.out.println("MySQL JDBC driver
not found!");
            } catch (SQLException e) {
```

```
                    System.out.println("Error: " +
e.getMessage());
                }
            }
        }
```

Output:

```
Enter student ID: 143
Enter student name: sri
Enter student age: 20
Student record inserted successfully
```

3. Write a JDBC program to fetch and display all student records from the database.

```
package Assesement_day11;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

public class Display_all_students_records {

    public static void main(String[] args) {

        try {


                Class.forName("com.mysql.cj.jdbc.Driver");
```

```
                Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/m
ydb", "root", "root");


                Statement stmt = con.createStatement();
                ResultSet rs = stmt.executeQuery("SELECT
* FROM student");
                while (rs.next()) {
                    System.out.println(rs.getInt(1) + " " +
rs.getString(2));
                }


                rs.close();
                stmt.close();
                con.close();
            } catch (ClassNotFoundException e) {
                System.out.println("MySQL JDBC driver
not found!");
            } catch (SQLException e) {
                System.out.println("Error: " +
e.getMessage());
            }
        }
```

```
}
```

Output:

```
143 sri
144 sanjana
145 dhana
```

4. Develop a program to search a student by ID using JDBC.

```
package jdbc_connectivity;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.Scanner;

public class Search_stuID {


    public static void main(String[] args) {

        String dbUrl = "jdbc:mysql://localhost:3306/mydb";

        String username = "root";

        String password = "root";
```

```
        try (Connection conn =
DriverManager.getConnection(dbUrl, username, password)) {
            Scanner scanner = new Scanner(System.in);
            System.out.print("Enter student ID: ");
            int studentId = scanner.nextInt();
            scanner.close();

            String query = "SELECT * FROM student WHERE
id = ?";
            try (PreparedStatement pstmt =
conn.prepareStatement(query)) {
                pstmt.setInt(1, studentId);
                try (ResultSet result =
pstmt.executeQuery()) {
                    if (result.next()) {
                        System.out.println("Student
found:");
                        System.out.println("ID: " +
result.getInt("id"));
                        System.out.println("Name: " +
result.getString("name"));
                        System.out.println("Age: " +
result.getInt("age"));
                    } else {
```

```
                                    System.out.println("Student not
found.");
                                }
                            }
                        }
                } catch (SQLException e) {
                        System.out.println("Error: " + e.getMessage());
                }
        }
}
```

Output:

```
Enter student ID: 143
Student found:
ID: 143
Name: sri
Age: 20
```

5. Write a Java program to delete a student record from the database using JDBC.

```
package Assesement_day11;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Scanner;
public class delete_student_record {
```

```
public static void main(String[] args) {
String dbUrl = "jdbc:mysql://localhost:3306/mydb";
String dbUser = "root";
String dbPassword = "root";

try {
Class.forName("com.mysql.cj.jdbc.Driver");

Connection conn = DriverManager.getConnection(dbUrl,
dbUser, dbPassword);

Scanner scanner = new Scanner(System.in);
System.out.print("Enter the student ID to delete ");
int studentId = scanner.nextInt();
String query = "DELETE FROM students WHERE id = ?";
PreparedStatement pstmt = conn.prepareStatement(query);
pstmt.setInt(1, studentId);

int rowsAffected = pstmt.executeUpdate();

if (rowsAffected > 0) {
System.out.println("Student record deleted successfully");
} else {
System.out.println("No student record found with the given
ID");
}

conn.close();
} catch (ClassNotFoundException e) {
System.out.println("MySQL JDBC driver not found.");
```

```
} catch (SQLException e) {
System.out.println("Error connecting to the database or
executing query: " + e.getMessage());
}

}

}
```

Output:

```
Enter the student ID to delete: 143
Student record deleted successfully
```

6.Design a Java application to perform all CRUD (Create, Read, Update, Delete) operations on an **Employee** table using JDBC.

```
package Assesement_day11;
import java.sql.*;
public class all_CRUD {

public static void main(String[] args) {
try {
Class.forName("com.mysql.cj.jdbc.Driver");

Connection conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/m
ydb", "root", "root");

createEmployee(conn, 1, "Sanjana", "java developer", 000);
```

```
readEmployees(conn);

updateEmployee(conn, 1, 60000);

readEmployees(conn);

deleteEmployee(conn, 1);

conn.close();
} catch (ClassNotFoundException | SQLException e) {
e.printStackTrace();
}
}

// Create employee
public static void createEmployee(Connection conn, int id,
String name, String position, double salary) throws
SQLException {
String query = "INSERT INTO employees (id, name, position,
salary) VALUES (?, ?, ?, ?)";
PreparedStatement pstmt = conn.prepareStatement(query);
pstmt.setInt(1, id);
pstmt.setString(2, name);
pstmt.setString(3, position);
pstmt.setDouble(4, salary);
pstmt.executeUpdate();
System.out.println("Employee created successfully");
}

// Read employees
```

```
public static void readEmployees(Connection conn) throws
SQLException {
String query = "SELECT * FROM employees";
Statement stmt = conn.createStatement();
ResultSet rs = stmt.executeQuery(query);
while (rs.next()) {
System.out.println("ID: " + rs.getInt("id"));
System.out.println("Name: " + rs.getString("name"));
System.out.println("Position: " + rs.getString("position"));
System.out.println("Salary: " + rs.getDouble("salary"));
System.out.println();
}
}

// Update employee
public static void updateEmployee(Connection conn, int id,
double salary) throws SQLException {
String query = "UPDATE employees SET salary = ? WHERE id
= ?";
PreparedStatement pstmt = conn.prepareStatement(query);
pstmt.setDouble(1, salary);
pstmt.setInt(2, id);
pstmt.executeUpdate();
System.out.println("Employee updated successfully");
}

// Delete employee
public static void deleteEmployee(Connection conn, int id)
throws SQLException {
String query = "DELETE FROM employees WHERE id = ?";
PreparedStatement pstmt = conn.prepareStatement(query);
```

```
pstmt.setInt(1, id);
pstmt.executeUpdate();
System.out.println("Employee deleted successfully");
}
}
```

Output:

```
Employee created successfully
ID: 1
Name: Sanjana
Position: java developer
Salary: 25000.0

Employee updated successfully
ID: 1
Name: Sanjana
Position: java developer
Salary: 60000.0

Employee deleted successfully
```

7. Create a JDBC-based program to count the total number of rows in a table.

```
package Assesement_day11;
import java.sql.*;
public class count_rows {

public static void main(String[] args) {
try {
Class.forName("com.mysql.cj.jdbc.Driver");
```

```
Connection conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/m
ydb", "root", "root");

Statement stmt = conn.createStatement();

ResultSet rs = stmt.executeQuery("SELECT COUNT(*) FROM
students");

if (rs.next()) {
int rowCount = rs.getInt(1);
System.out.println("Total number of rows: " + rowCount);
}

conn.close();
} catch (ClassNotFoundException | SQLException e) {
e.printStackTrace();
}
}
}
```

Output:

```
Total number of rows: 3
```

8. Develop a program to sort student data in ascending order by name using SQL in JDBC.

```
package Assesement_day11;
import java.sql.*;

public class Sort_student_data {
```

```
public static void main(String[] args) {
try {
Class.forName("com.mysql.cj.jdbc.Driver");

Connection conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/mydb", "root", "root");

String query = "SELECT * FROM students ORDER BY name";
Statement stmt = conn.createStatement();
ResultSet rs = stmt.executeQuery(query);

while (rs.next()) {
System.out.println("Name: " + rs.getString("name"));
}
conn.close();
} catch (ClassNotFoundException | SQLException e) {
e.printStackTrace();
}
}
}
```

Output:

```
Name: dhana
Name: sanjana
Name: sri
```

9.Use **PreparedStatement** to insert multiple student records into the database.

```
package Assesement_day11;
import java.sql.*;
public class insert_multiple_records {

public static void main(String[] args) {
try {
Class.forName("com.mysql.cj.jdbc.Driver");

Connection conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/m
ydb", "root", "root");

String query = "INSERT INTO students (name, grade, age)
VALUES (?, ?, ?)";
PreparedStatement pstmt = conn.prepareStatement(query);

insertStudent(pstmt, "John Doe", "A", 20);
insertStudent(pstmt, "Jane Doe", "B", 21);
insertStudent(pstmt, "Bob Smith", "A", 19);

conn.close();
} catch (ClassNotFoundException | SQLException e) {
e.printStackTrace();
}
}

private static void insertStudent(PreparedStatement pstmt,
String name, String grade, int age) throws SQLException {
pstmt.setString(1, name);
pstmt.setString(2, grade);
pstmt.setInt(3, age);
```

```
pstmt.executeUpdate();
System.out.println("Student record inserted successfully");
}

}
```

Output:

```
Student record inserted successfully
Student record inserted successfully
Student record inserted successfully
```

10.Write a JDBC program to handle exceptions (like invalid ID, connection errors) gracefully.

```
package Assesement_day11;
import java.sql.*;
public class handle_expections {

public static void main(String[] args) {
try {
Class.forName("com.mysql.cj.jdbc.Driver");

Connection conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/m
ydb", "root", "root");

Statement stmt = conn.createStatement();

try {
ResultSet rs = stmt.executeQuery("SELECT * FROM students
WHERE id = 143");
```

```
if (rs.next()) {
System.out.println("Student ID: " + rs.getInt("id"));
System.out.println("Student Name: " + rs.getString("name"));
} else {
System.out.println("No student record found with the given
ID.");
}
} catch (SQLException e) {
System.out.println("Error executing query: " + e.getMessage());
} finally {
try {
stmt.close();
} catch (SQLException e) {
System.out.println("Error closing Statement object: " +
e.getMessage());
}
}
} catch (ClassNotFoundException e) {
System.out.println("MySQL JDBC driver not found");
} catch (SQLException e) {
System.out.println("Error connecting to the database: " +
e.getMessage());
}

}

}
```

Output:

Student ID: 143
Student Name: sri